

Verificación de la integridad de archivos mediante funciones hash



Índice

Enunciado de la tarea.....	3
Resolución de la tarea.....	5
Práctica el Windows.....	5
Práctica en Linux.....	7
Comprobación de algoritmos.....	8
Informe final.....	8

Enunciado de la tarea

Objetivos:

El objetivo de esta tarea es que los estudiantes comprendan el uso de las funciones hash para garantizar la integridad de los archivos, tanto en sistemas Windows como Linux. Aprenderán a generar y verificar huellas digitales utilizando distintas herramientas y algoritmos de hash, reforzando el concepto de seguridad e integridad en los sistemas.

Instrucciones:

1. Práctica en Windows:

Utilizando la herramienta CertUtil en Windows, calcula el hash de un archivo que tengas disponible en tu ordenador (puede ser un archivo de texto pequeño o cualquier archivo que elijas).

Usa al menos dos algoritmos diferentes (ej. MD5 y SHA256).

Copia y pega en el informe los comandos utilizados y los resultados obtenidos.

Opcional: Instala HashTab o QuickHash GUI y genera el hash de un archivo usando uno de estos programas. Explica brevemente tu experiencia con estas herramientas (¿te parecieron fáciles de usar?, ¿cuál es la ventaja de tener una interfaz gráfica?).

Haz captura del uso de la herramientas

2. Práctica en Linux:

Utilizando una máquina virtual con Linux o una distribución en modo live, calcula el hash de un archivo usando los siguientes comandos:

md5sum para generar el hash MD5.

sha256sum para generar el hash SHA-256.

Copia y pega en el informe los comandos utilizados y los resultados obtenidos.

Explica cómo podrías utilizar estos comandos para verificar la integridad de un archivo descargado de internet.

3. Comparación de Algoritmos (análisis):

Investiga y responde: ¿Por qué los algoritmos MD5 y SHA-1 ya no son recomendados para aplicaciones críticas? Da un ejemplo de una situación en la que el uso de estos algoritmos podría representar un riesgo.

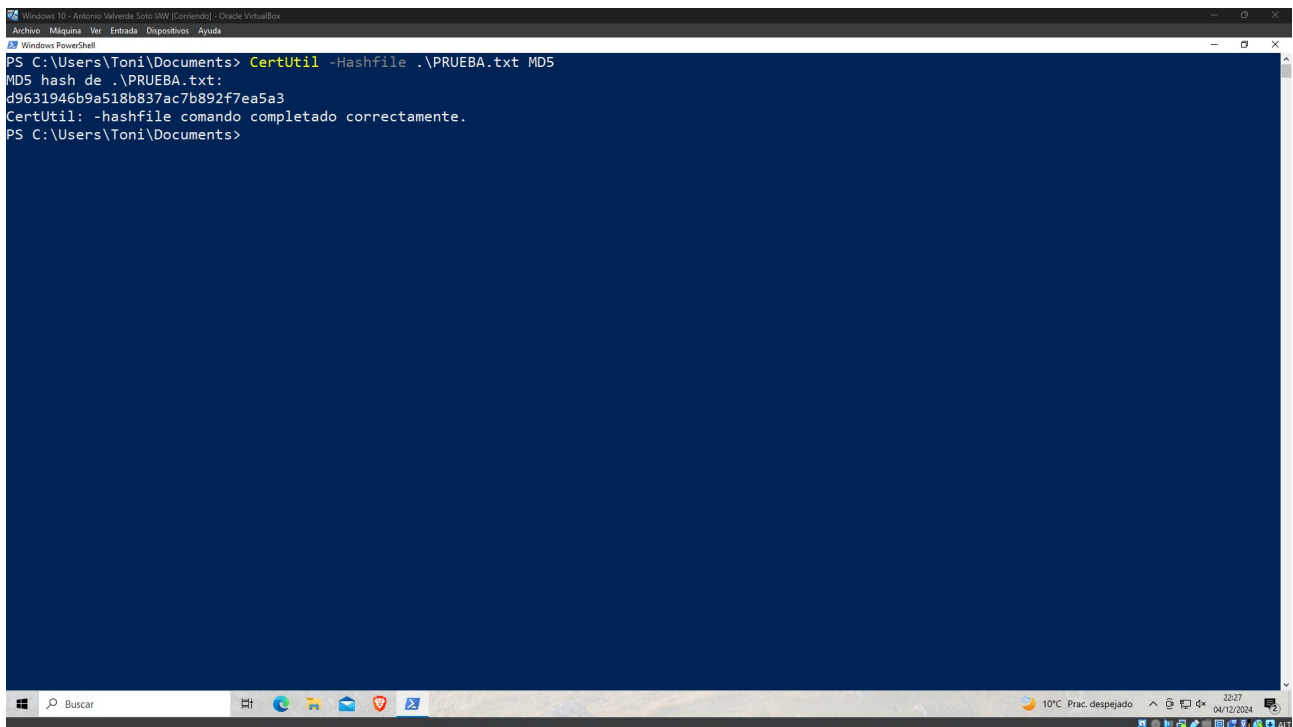
Indica en qué situaciones podría ser aceptable utilizar MD5 en lugar de algoritmos más seguros como SHA-256 o SHA-512.

4. Informe Final:

Elabora un informe en un documento de texto (.docx o .pdf) en el que incluyas las respuestas a las preguntas anteriores, capturas de pantalla de la ejecución de los comandos y una breve reflexión sobre la importancia de las funciones hash en la seguridad informática.

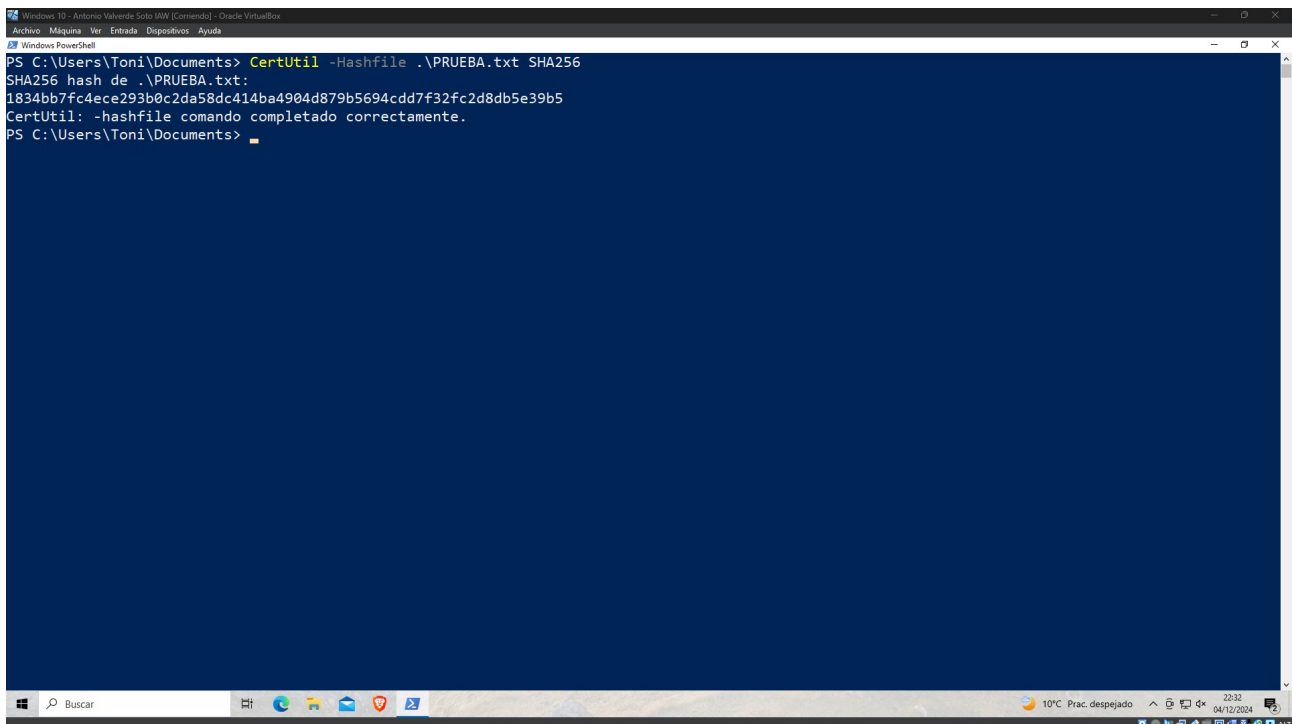
Resolución de la tarea

Práctica el Windows



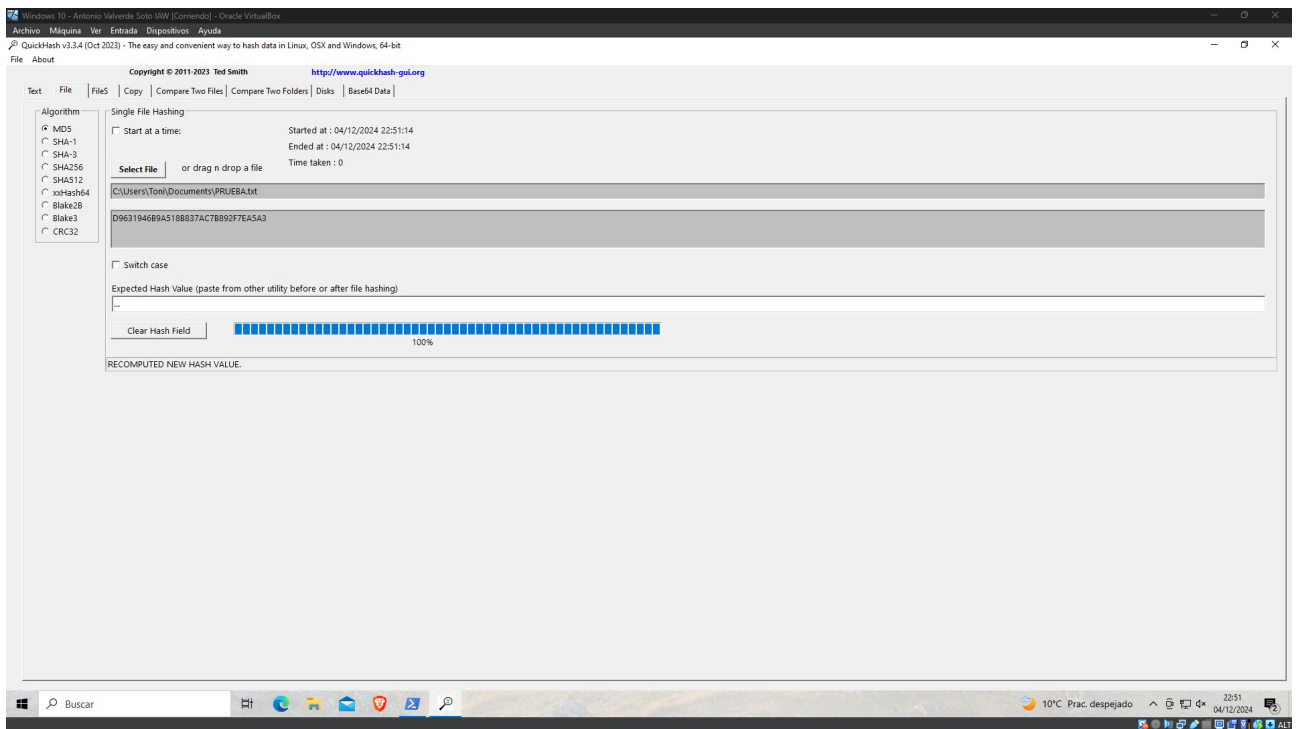
```
Windows 10 - Antonio Valverde Soto VMW [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Windows PowerShell
PS C:\Users\Toni\Documents> CertUtil -Hashfile .\PRUEBA.txt MD5
MD5 hash de .\PRUEBA.txt:
d9631946b9a518b837ac7b892f7ea5a3
CertUtil: -hashfile comando completado correctamente.
PS C:\Users\Toni\Documents>
```

Con el uso del comando CertUtil -Hashfile podemos comprobar el hash de un algoritmo en concreto del fichero que deseemos. Esto es lo mostrado MD5 hash de .\PRUEBA.txt: d9631946b9a518b837ac7b892f7ea5a3

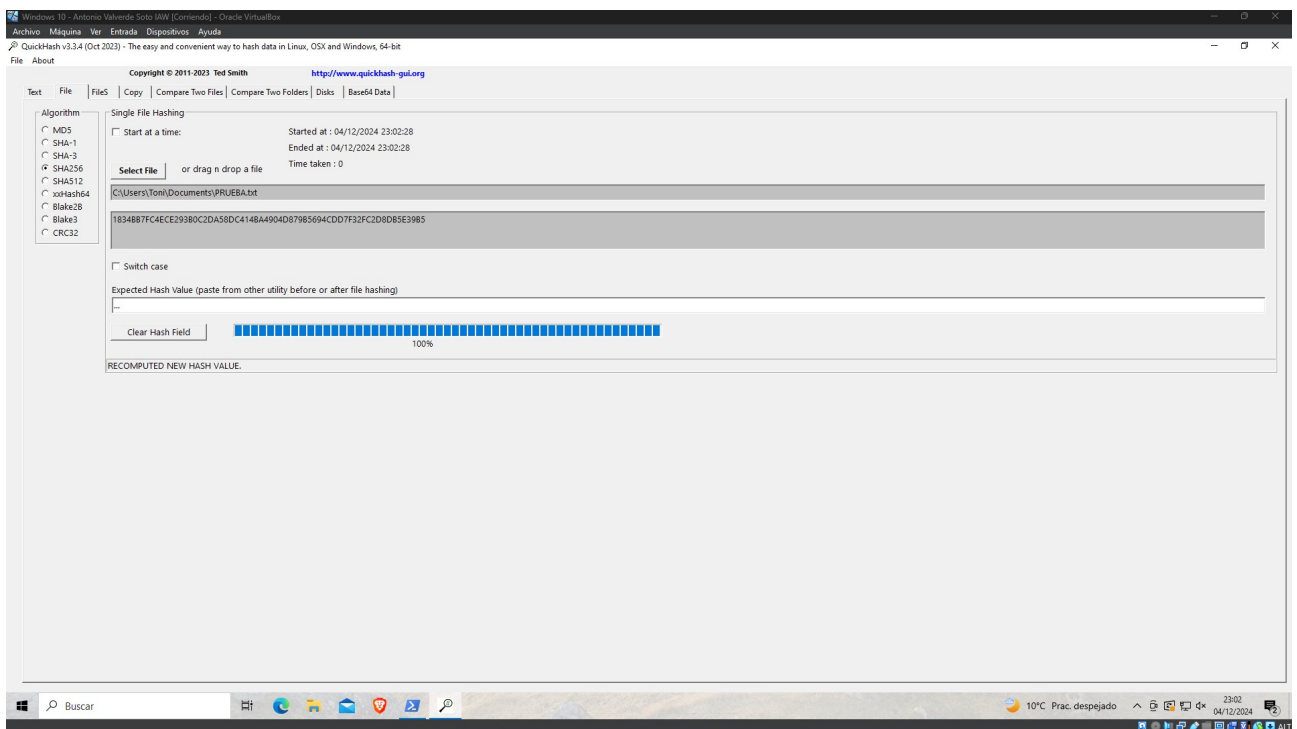


```
Windows 10 - Antonio Valverde Soto VMW [Corriendo] - Oracle VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Windows PowerShell
PS C:\Users\Toni\Documents> CertUtil -Hashfile .\PRUEBA.txt SHA256
SHA256 hash de .\PRUEBA.txt:
1834bb7fc4ece293b0c2da58dc414ba4904d879b5694cdd7f32fc2d8db5e39b5
CertUtil: -hashfile comando completado correctamente.
PS C:\Users\Toni\Documents>
```

Si hacemos lo mismo pero con sha256, obtendremos:SHA256 hash de .\PRUEBA.txt: 1834bb7fc4ece293b0c2da58dc414ba4904d879b5694cdd7f32fc2d8db5e39b5



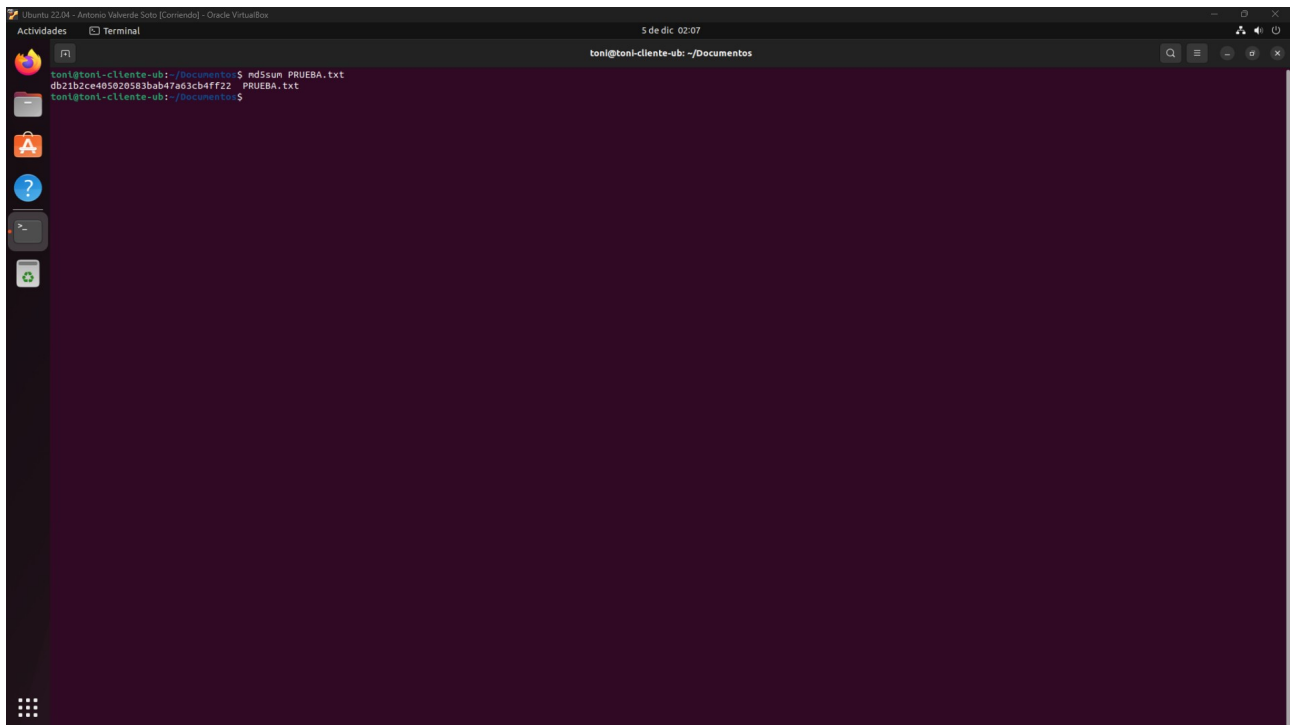
Usando QuickHash Gui facilitamos mucho la obtención del hash, ya que únicamente tenemos que seleccionar el fichero y el algoritmo y automáticamente veremos el hash.



Para poder ver el hash del algoritmo sha256, simplemente bastará con seleccionar el nuevo algoritmo y el hash se sustituirá por el que queremos.

Esta herramienta es muy sencilla de utilizar, gracias a la posibilidad de tener el entorno gráfico, ya que gracias a él, únicamente con 3 clics podremos acceder al hash del algoritmo que queramos.

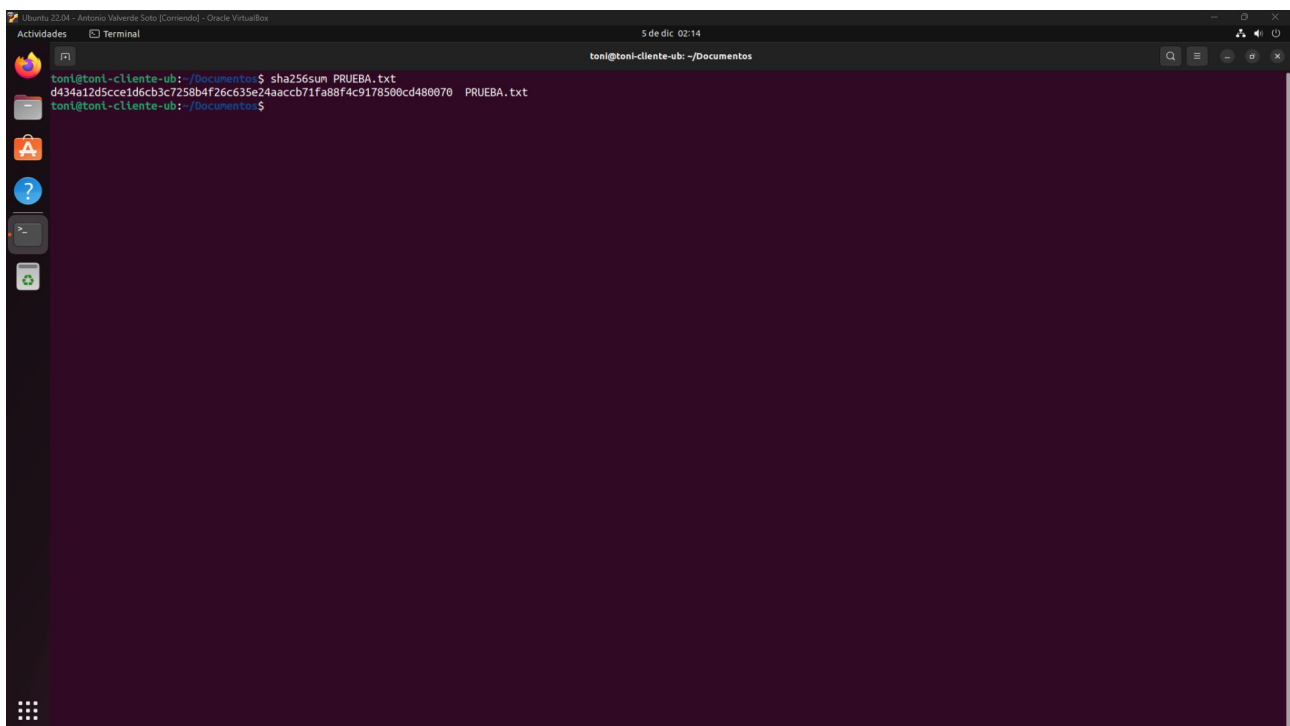
Práctica en Linux

A screenshot of a Linux terminal window. The window title is 'Ubuntu 22.04 - Antonio Valverde Soto [Comando] - Oracle VM VirtualBox'. The terminal shows the user 'toni' at the prompt 'toni@toni-cliente-ub: ~/Documentos'. The command 'md5sum PRUEBA.txt' has been executed, resulting in the output 'db21b2ce405020583bab47a63cb4ff22 PRUEBA.txt'.

```
toni@toni-cliente-ub: ~/Documentos$ md5sum PRUEBA.txt
db21b2ce405020583bab47a63cb4ff22 PRUEBA.txt
toni@toni-cliente-ub: ~/Documentos$
```

Usando el comando md5sum PRUEBA.txt, podremos ver lo siguiente:

db21b2ce405020583bab47a63cb4ff22 PRUEBA.txt

A screenshot of a Linux terminal window. The window title is 'Ubuntu 22.04 - Antonio Valverde Soto [Comando] - Oracle VM VirtualBox'. The terminal shows the user 'toni' at the prompt 'toni@toni-cliente-ub: ~/Documentos'. The command 'sha256sum PRUEBA.txt' has been executed, resulting in the output 'd434a12d5cce1d6cb3c7258b4f266c635e24aacb71fa88f4c9178500cd480070 PRUEBA.txt'.

```
toni@toni-cliente-ub: ~/Documentos$ sha256sum PRUEBA.txt
d434a12d5cce1d6cb3c7258b4f266c635e24aacb71fa88f4c9178500cd480070 PRUEBA.txt
toni@toni-cliente-ub: ~/Documentos$
```

Esta vez, usamos el comando sha256sum PRUEBA.txt y hemos obtenido lo siguiente:

d434a12d5cce1d6cb3c7258b4f266c635e24aacb71fa88f4c9178500cd480070
PRUEBA.txt.

Si queremos utilizar este comando para comprobar la integridad de un archivo descargado desde internet, por ejemplo de una iso, usaremos este comando, y en internet, en la página oficial desde la que se descarga la iso comprobaremos un hash con otro.

Comprobación de algoritmos

MD5 y SHA-1 no son recomendadas ya a día de hoy y están en desuso debido a las grandes vulnerabilidades que éstas presentan, tienen vulnerabilidades de colisión, con el uso de avances en el poder computacional y sistemas de firma digital y certificación, estos entre otros, ya que éstos son los que yo he encontrado, y cada día que pasa se presentan más vulnerabilidades, por lo que no es ni mucho menos recomendable usarlos, ya que por ejemplo en el caso de las firmas digitales, el atacante podría crear una versión maliciosa del archivo produciendo el mismo hash (siendo éste del algoritmo sha-1 por ejemplo), por lo cuál habría un gran problema.

Informe final

La comprobación del hash es muy importante realizarlo siempre que se pueda en un entorno seguro, ya que de esa forma podremos prevenir que accedan a nuestro contenido o nos ataquen a través de ese fichero, antes he puesto el ejemplo de las isos, pero lo voy a volver a usar, debido tanto a su fácil comprobación, como a los grandes peligros que puede presentar si ha sido modificado. Desde la propia página desde dónde instalamos la iso de instalación, debemos ser capaces de encontrar el hash de un algoritmo, y con las herramientas mostradas anteriormente, podríamos comprobar la integridad del archivo.